Part1

Hello, my name is _____,

Thank you _____ for participating in our study. We aim to gather insights into testing for and measuring discriminatory behavior in the outcome of software systems. We look forward to utilizing your insights to further our research.
Now, we will review the consent form included in the interview sign-up survey.

[Revisit the consent form and open for questions]

This interview will be recorded for internal purposes only. Additionally, we will be taking notes during the interview. The consent form that was sent to you specifies that your identity will remain confidential. Other than your email address, no other personal information will be collected. Your email will be used only to schedule your interview and compensation. You can withdraw the study at any point. Are you still willing to participate in this study?

If you get disconnected from the call, please use the same link included in the email invitation, join via phone, or email if you need to reschedule.

Before we begin, do you have any questions for me?
I'll begin the interview recording now.
[Start Recording]

Background Questions

1. To begin, tell us a little about yourself.
2. What is your current role/occupation?
    a. How familiar are you with traditional software testing methodologies? Maybe you can share the kind of experience in software testing with us.

Interview Questions

Thank you for responding! Now I want to talk more about your experiences & expectations in understanding discriminatory behavior in software systems.

********** **Part 1**********

Section: Experiences with Discrimination in Software Outcomes

1. Can you describe an experience encountering or thinking about discrimination in software outcomes?
    - What type of software system was it?
    - How would you describe your overall experience in that situation?

    - When you recognized that potential for discriminatory behavior, what was your *expectation* of how the system should have behaved instead?

2. Prior work identifies five major types of discrimination in software outcomes: **Algorithmic Bias, Technological Bias, Demographic Bias, Accessibility Bias,** and **Other Human Values.**

    - Which of these categories do you feel best represents your experience?

    - Have you experienced or thought about any other categories of software discrimination?

3. Do you think identifying and reporting potentially discriminatory behavior is part of a software tester's responsibility?

    - Why or why not?

6. Have you ever used any tools or frameworks to detect bias or discrimination in software systems?

    - If yes, which tools?

    - If not, do you rely on your own implementation or manual checks?

7. Based on your experience, have you encountered community documentation or discussions (e.g., in project documentation, GitHub issues, discussion forums, research articles) related to discrimination/inclusivity(for example)?

    - If so, where have you observed these discussions taking place?

Part2

*******Part 2********

Thank you for sharing your experiences and perceptions regarding software discrimination. Next, we would like to get your perception on the utility and usability of different approaches to create unit tests for measuring and understanding discriminatory behavior in software outcomes. For this study we are considering two different test generation tools. First is Pynguin which is an automated unit test generation for Python. Second is Themis, an automated causal unit test generation tool.

Now I will elaborate for you 2 separate scenarios. I want you to open our project and do the following tasks according to instructions, which you will have access to as you are working

Section 1: Test Run

Scenario 1: This is a simple loan application system which is a Rule-based software in slide 4. The behavior of this software is quite deterministic. We aim to understand the user perspective on understanding discriminatory behavior of such systems through unit test generation approaches.

- **Input Specification Task**

A. Scenario: Given the loan approval case, participants define input attributes (e.g., gender, race, income) as categorical.

- **Think-Aloud Prompt** (generated testcases using both **Pynguin & Themis**)

1. What do you think these unit tests are verifying/communicating?

2. Do you think the generated unit tests help you reason about how the program treats different groups?
    - If so, why?
    - If not, why not?

3. Based on these results, do you notice any potential discriminatory pattern in this software's outcome? *(Open-ended)*

4. How could these two types of tests complement each other in evaluating software inclusivity? *(Open-ended)*

5. What type of additional information would help you understand discrimination more clearly from these tests?

Section 2— Comparison Deterministic vs Non deterministic(only for Themis):

Scenario 2: In this next scenario, we will test a Telecom Churn prediction software that uses DL models. We want users to understand discrimination in such a non deterministic DL-based software outcome**.**

- **Input Specification Task**

A. Scenario: Given churn prediction dataset and as the input subspace is huge we aimed to use a subset of input space, participants define input attributes (e.g., gender, SeniorCitizen, partner, dependents, tenure) as categorical and other test configurations are similar to loan application software.

B. Generate automated test suite through the tool you preferred in earlier use case for understanding bias

- **Think-Aloud Prompt**

What do you think these test cases are designed to verify about the DL model(in slide 13 and 15)? *(Open-ended) in*

From these tests, can you infer any patterns of bias or discrimination? *(Open-ended)*

Q1. What do you find useful about Themis's approach of generating repetitive testcase for non-deterministic systems(DL Churn Prediction), where it may produce different results for the same test case?*(open ended)*

Q2. How can or should discrimination-oriented testing methodology differ between rule-based and DL-based decision making systems?*(Open-ended)*

Q3. Did these generated tests help you to decide to make any changes to the program/dataset to mitigate discrimination in outcome? If yes, How so?*(Open-ended)*

Q4. Do you find altering more than one attribute can be more insightful? Do you think altering more than one value can provide more deeper insight in understanding causal discrimination.

Q5. Do you have any recommendations for improving causal testing methodology for understanding discrimination in software outcomes?

Part3

******Part3******

LLM has been identified as a promising tool for unit test generation. We want to identify if LLM can play a role in such discrimination testing. For this we considered ChatGPT. Use the loan application software.

Prompt: We will keep it open for the user to write prompts for analyzing such kinds of discrimination.

- What strengths or limitations did you notice in GPT's discrimination reasoning compared to Themis? *(Open ended)*
- What additional information or structure would make GPT's causal discrimination reasoning more useful or trustworthy?
- What are the possible challenges we need to overcome to utilize LLM for discrimination tests effectively?